On Ontologogy

On Ontologology

Perdita Stevens* Jeremy Gibbons†

* School of Informatics
University of Edinburgh, UK
Perdita.Stevens@ed.ac.uk

† Department of Computer Science
University of Oxford, UK
Jeremy.Gibbons@cs.ox.ac.uk

Abstract

The study of models, and related concepts such as metamodels, is largely situated within the software engineering community under the banner of model-driven development. Yet these concepts have some obvious parallels with concepts developed within the artificial intelligence community under the banners of ontologies and the semantic web. Although a considerable amount of work has been done that aims to relate the development of ontologies to the model-driven development of software, the place of bidirectional transformations within these connected worlds is (almost) unstudied. Yet, experts in the study of ontologies have experienced the need to check and restore consistency, and have developed techniques, terminology and tools that relate to these tasks. In this paper we provide a high-level introduction to the work that has been done, aiming to promote further study and perhaps collaboration between these communities.

1 Introduction

The intended reader of this short paper is a member of the bx community who is – like the authors not long ago – aware of the existence of a community to which "ontology" is a key term, and suspects that its work may have something to do with their own, yet who is not familiar with the ontologological literature. We aim to provide some pointers into that literature, together with brief explanations of key terms. Since it turns out that a lot of work has been done on the relationship between ontologies, and semantic web technologies more generally, and model-driven engineering, we focus here on issues most connected to bx specifically.

Ontologies have been studied for a long time, by a large community, so we need to restrict our scope. We will focus on ontologies as used in the artificial intelligence community and on the kinds of ontologies that are *engineering artefacts*. For example, while we will mention ontology matching, we will not attempt to cover schema matching more broadly (we refer to [9] for a comparison); and we will not discuss ontologies that emerge [12] from a community's use of terms, either by social mechanisms such as tagging (folksonomies) or by the use of machine learning. We hope in this way to facilitate connections with the research that is the most relevant to the bx community without already being widely known within it.

2 What is an ontology?

According to a very old and highly-cited paper [8], an ontology is an *explicit specification of a conceptualization.* That is, it makes precise an abstract representation of the entities that are of interest to people working in some

In: R. Eramo, M. Johnson (eds.): Proceedings of the Sixth International Workshop on Bidirectional Transformations (Bx 2017), Uppsala, Sweden, April 29, 2017, published at http://ceur-ws.org

domain, and the relationships between them. We might say: an ontology is a model. More challenging for our community are definitions such as this from `http://www.ontologymatching.org/`: *An ontology typically provides a vocabulary that describes a domain of interest and a specification of the meaning of terms used in the vocabulary.* The "vocabulary" part is straightforward, but what of "meaning"? To ontologists, meaning is captured in two ways. Formally, an ontology is provided with a collection of axioms. These are comparable to the constraints that we often place on models, but typically, a greater emphasis is placed on reasoning with the axioms and they may be expected to be, in some sense, complete. Informally, an ontology must be meaningful in the sense that it must correctly express facts about the real world. Ontologists may talk about a correct deduction from an ontology's axioms being "false", by which they mean that it represents a statement about the real world which is false there.

The formal-minded reader should keep in mind the following definition, quoted verbatim from [4]:

Definition *An ontology is a pair* (Σ, A)*, where:*

- Σ *is the ontology's signature, which defines its language. It consists of a set of type declarations for the concepts in the ontology.*
- A *is the ontology's axioms, which define its theory. It consists of a set of formulae asserted to be true.*

The type declarations and formulae are expressed in a logic $\mathcal{L}$.

Much has been written on the relationship between ontologies and model-driven development, and we will not attempt to recapitulate it here; good starting points are [7], [1] and indeed the rest of the book [6] in which the latter appears. For most purposes the development of an ontology parallels that of a computation independent model (CIM) in OMG MDA terms; complications arise from ontology's lack of a clean separation between metalevels. OMG has standardised an Ontology Definition Metamodel [13] which facilitates relating models and ontologies and is used in several of the papers cited here.

The cited authors and others have attempted to make distinctions between models and ontologies, but some (descriptive ontologies vs prescriptive models; ontologies with an open world assumption, models with a closed world assumption) look dated from the viewpoint of modern, increasingly agile, MDD, in which models are often used to describe, and in the knowledge that they are incomplete. In fact, similar concerns often arise in both settings. Of course assumptions – closed world versus open world, especially – underlying logical reasoning do have to be made with care, and this is important because one of the main reasons for MDD practitioners to be interested in ontology is that there are well-established facilities for formal reasoning, with good tool support. Wagelaar, for example, discusses such issues [21] in the context of incorporating OWL DL expressions in ATL transformations, in order to take advantage of facilities for reasoning over OWL DL.

Both bx and ontology communities talk about "consistency" and this can lead to confusion; while we mean the notion of consistency defined by the bx writer, they mean logical consistency of a set of axioms. Arguably the connection is: the bx writer's consistency must be equivalent to the logical consistency of *unwritten* axioms connecting the models to the future real world that includes the eventual implemented system.

3 Matching and alignment

Ontology matching, also known as *ontology alignment*, is the process of relating two ontologies in order that they can be used simultaneously in some application; for example, so that a query can be posed to data sources described by the two ontologies and can retrieve comparable results. Implicitly, the correctness of an alignment is given by the fact that each ontology has semantics in the real world. In the very useful overview paper [18] by Shvaiko and Euzanat, ontology matching is described as *a solution to the semantic hetereogeneity problem*, and this is echoed in `http://www.ontologymatching.org/`: *Ontology matching is a promising solution to the semantic heterogeneity problem. It finds correspondences between semantically related entities of the ontologies. These correspondences can be used for various tasks, such as ontology merging, query answering, data translation, or for navigation on the semantic web. Thus, matching ontologies enables the knowledge and data expressed in the matched ontologies to interoperate.*

The study of ontology matching, and tool support for it, is mature (e.g. compared with that of model merging). An annual contest for ontology matching tools, the Ontology Alignment Evaluation Initiative (OAEI)[1], has taken place each year since 2004. Successful tools typically make heavy use of various kinds of string comparison, but

[1] `http://oaei.ontologymatching.org`

may also use structural features of the ontologies and external sources of information (e.g. WordNet for identifying synonyms). In bx terms we may see these tools as inferring a consistency relation between model sets, making heavy use of outside information. This is one place where the bx community may be able to learn from the ontology community.

Some authors have, indeed, begun to use ontology matching techniques for tasks in MDD. For example, Kappel et al. in [10] propose to ease the integration of modelling tools based on different metamodels by what they term "lifting" the metamodels to ontologies, which they then match. The key advantages of doing so seem to be (a) abstraction, in that the process enables the key domain concepts to be separated from more technical metamodel classes (b) the availability of reasoners on the ontology level. The process used here is largely manual, however, and it is not clear how the cost/benefit compares with other ways of solving the same problem. Issues of bidirectionality are not explicitly addressed; it may be that the main problems targetted are where the bx consistency relation is bijective.

For all the maturity of the field of ontology matching, some key concerns familiar to the bx community do not seem to have been fully addressed, and this is where the ontology community might also benefit from collaboration. In the terminology of [18] an alignment contains correspondences such as $\langle id_7, \texttt{Book}, \texttt{Monograph}, \sqsubseteq \rangle$, indicating that `Book` in one ontology is more general than `Monograph` in the other. Software using the alignment expects to be able to transform a monograph into a book, but not necessarily vice versa. What seems understudied in ontologies is the – usual! – case where neither concept is more general than the other, but they overlap more subtly; that is, where there is a bx between the concepts! Thus, this seems likely to be a field where the bx community could both learn and contribute; e.g. connecting with witness structures, and with experience in recursive and conditional matching (cf QVT-R's `when` and `where`). It is interesting to note the concern in [18] that tools should be able to *explain* their alignment decisions; the present authors have recently begun to think about explanations in the context of bx.

A complicating factor, for the bx reader trying to understand ontology matching, is that several different tasks are included under that heading, depending on what needs to be done with the two ontologies that are to be matched. Besides the static view mentioned, in which the task is to link concepts in one ontology with concepts in the other, some work labelled ontology matching is concerned with translating data from one ontology to another. Alternatively, the latter task is sometimes called *ontology translation*, with an ontology alignment or matching seen as a specification of this translation. It is possible to see the alignment task as corresponding to a bx's specification of consistency – or to inferring such a notion of consistency – and the translation task as corresponding to a bx's enforcement of consistency; but the match is not exact, for example because there seems to be no aim in the ontology community to build a single artefact capturing both consistency and its bidirectional restoration.

In [14] translation is done unidirectionally using a language based on ATL, but the lack of bidirectionality is explicitly mentioned as a limitation. In [15] a limited amount of bidirectionality is possible, in cases where, effectively, the bx consistency relation can be expressed in terms of bijections between sets of individuals; this might repay further study.

4 Repair

Ontology repair is defined in [3] in terms of the relationship between an ontology and the real world. Informally, an ontology repair modifies an ontology, and induces a modification of any sentence about the ontology. Such a modification is a repair if, whenever a sentence is entailed by the axioms of the ontology but false in the real world (that is, "a false statement can be proved"), then either in the modified ontology the modified sentence is not entailed, or else, the modified sentence is true in the real world. This definition is notable for being all or nothing: we repair the ontology all the way to perfection in one step.

Later work looks at the repair process at smaller granularity. In [4], one kind of *fault* in an ontology is a "false" statement that can be proved. To *diagnose* a fault in an ontology is to find a pair of a signature morphism and a theory morphism that can be applied to the ontology, after which the false statement can no longer be proved; to *repair* the fault is simply to apply these morphisms. In this sense, a repair of an ontology does not necessarily result in a faultless ontology. It might be an improvement in the sense of [19] but we are not aware of literature on this, nor on least change concerns.

Some work has been done on supporting ontology repair by tools. The GALILEO system [5], for example, makes the connection between repair and bx more obvious. It compares two ontologies, one representing physicists' theoretical understanding, and one representing experiments together with their conclusions. One kind of fault in the theoretical ontology is a statement provable within it which is false in the experimental ontology. Bundy and Chan discuss how to use the proof tree of the false statement to identify the problem, which may be a false axiom or a problem in the ontology's language such as a missing argument to a predicate, and how to repair it. (The possibility of changing the language, not just the theory, is the key difference between this work and the related field of *belief revision*.) The work relies on the ability to use the other ontology as an oracle, which can say when a statement is false, provided the statement is expressible in its language. One example from [5] concerns Joseph Black's use of a freezing experiment to separate the concepts of heat and temperature: the authors describe how to model the resolution of the problem by formally splitting one concept into two.

Clearly the bx community is considering similar ideas, and perhaps collaboration may benefit both; however, the precise connections are unclear at this stage. A key issue is the extent to which it makes sense to think of repairing an ontology, with respect to the real world, as restoring consistency between it and the real world (including by acting on the world).

Ontology update is a related concept that focuses on providing a means to *express* a semantically sensible change that should be made to an ontology (as opposed to considering whether the change *improves* the ontology). In [11] a distinction is made between changes that are motivated by the discovery of a problem with an ontology – such as ontology repairs – and those motivated by changes in the real world that the ontology is supposed to reflect. For the latter, an architecture and a supporting Ontology Update Language is proposed. We may think of these as providing an edit language for the ontology, incorporating domain knowledge about which edits are sensible. There are intriguing connections with least change considerations (which space forbids elaborating here: essentially, the edit language can incorporate knowledge so that the repair can be done in a "better" way than by naive metric least change).

Later work has refined the implementation of such ontology updating, including the dynamic invocation of updates and their chaining; see for example [17] and `http://wiki.opensemanticframework.org/index.php/Ontology_Update`. Perhaps there are connections with delta-based bx that would repay study?

5 Related work

As indicated, the relationship between an ontology and a (meta)model has been explored by many authors. However, explicit connections with bidirectional transformations are almost absent from the literature. An exception is [20], in which Textor proposes the use of (classic) lenses to manage the relationship between an ontology that includes instances and an external data source that acts as a view. Here the ontology is effectively being used as a model space, assumed to include both type information for the model and at least one model instance. In this short paper, there is mention of some of the issues mentioned above, such as the problem of updates introducing inconsistency, but solutions are postponed to future work. In [16] the authors present a very ambitious plan, supported by a prototype tool, to generate and evolve model transformations (unidirectional in this case, but in QVT-R) by making use of a reference ontology to which both metamodels are mapped, with the help of ODM. In [2], TGG transformations are translated into SPARQL; however, since SPARQL is a unidirectional language, naturally the different modes of use of the TGG each have to be translated separately.

6 Conclusions

We have briefly pointed at some of the literature from ontologies that seems likely to be of interest to **Bx** participants, in the hope of supporting future cross-fertilisation.

Acknowledgements We thank Alan Bundy, Juan Casanova and Dragan Gasevic for helpful writings, pointers into the literature, and conversations. All misunderstandings are our own. We also thank the anonymous reviewers, whose helpful suggestions and pointers into the literature we have only begun here to assimilate: a full paper that will do these justice is work in progress.

References

[1] Uwe Aßmann, Steffen Zschaler, and Gerd Wagner. Ontologies, meta-models, and the model-driven paradigm. In Calero et al. [6], chapter 9, pages 249–273.

[2] Robert Bill, Simon Steyskal, Manuel Wimmer, and Gerti Kappel. On synergies between model transformations and semantic web technologies. In *Proceedings of the 8th Workshop on Multi-Paradigm Modeling (MPM@MODELS 2014)*, volume 1237 of *CEUR Workshop Proceedings*, pages 31–40. CEUR-WS.org, 2014.

[3] Alan Bundy. Towards a theory of ontology repair, or, truthfulness considered harmful. Presented at Automated Reasoning Workshop 2007.

[4] Alan Bundy. Towards a theory of diagnosis of faulty ontologies. In *Workshop on Discovering Meaning On the Go in Large Heterogeneous Data (LHD-11)*, pages 14–18, 2011.

[5] Alan Bundy and Michael Chan. Towards ontology evolution in physics. In Wilfrid Hodges and Ruy J. G. B. de Queiroz, editors, *WoLLIC 2008*, volume 5110 of *LNCS*, pages 98–110. Springer, 2008.

[6] Coral Calero, Francisco Ruiz, and Mario Piattini, editors. *Ontologies for Software Engineering and Software Technology.* Springer Berlin Heidelberg, 2006.

[7] Dragan Djuric, Dragan Gasevic, and Vladan Devedzic. Ontology modeling and MDA. *Journal of Object Technology*, 4(1):109–128, 2005.

[8] Thomas R. Gruber. A translation approach to portable ontology specifications. *Knowledge Acquisition*, 5(2):199–220, 1993.

[9] Petko Ivanov and Konrad Voigt. Schema, ontology and metamodel matching – Different, but indeed the same? In Ladjel Bellatreche and Filipe Mota Pinto, editors, *MEDI'11*, volume 6918 of *LNCS*, pages 18–30. Springer, 2011.

[10] Gerti Kappel, Elisabeth Kapsammer, Horst Kargl, Gerhard Kramler, Thomas Reiter, Werner Retschitzegger, Wieland Schwinger, and Manuel Wimmer. Lifting metamodels to ontologies: A step to the semantic integration of modeling languages. In *MODELS'06*, volume 4199 of *LNCS*, pages 528–542. Springer, 2006.

[11] Uta Lösch, Sebastian Rudolph, Denny Vrandecic, and Rudi Studer. Tempus fugit: Towards an ontology update language. In *ESWC'09*, pages 278–292, 2009.

[12] Peter Mika. Ontologies Are Us: A Unified Model of Social Networks and Semantics. In Yolanda Gil, Enrico Motta, V. Richard Benjamins, and Mark A. Musen, editors, *ISWC'05*, volume 3729 of *LNCS*, pages 522–536. Springer, November 2005.

[13] OMG. Ontology definition metamodel version 1.1. OMG document formal/2014-09-02, 2014. available from www.omg.org.

[14] Fernando Silva Parreiras, Steffen Staab, Simon Schenk, and Andreas Winter. Model driven specification of ontology translations. In *ER'08*, volume 5231 of *LNCS*, pages 484–497. Springer, 2008.

[15] Federico Rieckhof, Mirko Seifert, and Uwe Assmann. Ontology-based model synchronisation. In Fernando Silva Parreiras, Jeff Z. Pan, and Uwe Assmann, editors, *Proceedings of the 3rd Workshop on Transforming and Weaving Ontologies in Model Driven Engineering (TWOMDE-10)*, volume 604 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2010.

[16] Stephan Roser and Bernhard Bauer. Automatic generation and evolution of model transformations using ontology engineering space. *J. Data Semantics*, XI:32–64, 2008. Volume 5383 of *LNCS*.

[17] Jordy Sangers, Frederik Hogenboom, and Flavius Frasincar. Event-driven ontology updating. In X. Sean Wang, Isabel Cruz, Alex Delis, and Guangyan Huang, editors, *WISE'12*, pages 44–57. Springer, 2012.

[18] Pavel Shvaiko and Jérôme Euzenat. Ontology matching: State of the art and future challenges. *IEEE Trans. Knowl. Data Eng.*, 25(1):158–176, 2013.

[19] Perdita Stevens. Bidirectionally tolerating inconsistency: Partial transformations. In Stefania Gnesi and Arend Rensink, editors, *FASE'14*, volume 8411 of *LNCS*, pages 32–46. Springer, 2014.

[20] Andreas Textor. Bi-Directional Ontology Updates Using Lenses. In Dilletta Romana Cacciagrano and Petre Dini, editors, *SEMAPRO'12*, Barcelona, Spain, September 2012.

[21] Dennis Wagelaar. Towards using OWL DL as a metamodelling framework for ATL. In *Proceedings of the 2nd International Workshop on Model Transformation with ATL (MtATL-2010)*, volume 711 of *CEUR Workshop Proceedings*, pages 79–85. CEUR-WS.org, 2010.